Remarks

In the Office Action dated July 15, 2005 ("Office Action"), Claims 1, 2, 4-6, 12-24, 29-32 and 34-36 were rejected and Claims 3, 7-11, 25-28, 33 and 37 were objected to. In the amendment set forth above, Claims 1, 12, 14, 20, 24, 29, and 34 are amended, and the remaining Claims are as previously presented. In view of the amendments to the Claims and the comments set forth below, it is respectfully submitted that all Claims are now in condition for allowance as presently presented.

1. Claims 24 was objected to because it is said to be awkward. This Claim has been amended to correct a typographical error. With this change, it is believed the Examiner's objection to Claim 24 and the Claims that depend from Claim 24 has been addressed, and it is requested that this objection be withdrawn.

2. Claims 14-19 were rejected under 35 USC §112 as being indefinite for failing to particularly point out and distinctly claim the subject matter of the invention. In particular, the term "a request" in Claim 14 is said to be unclear because it is not clear to which request the term is referring. This phrase has been amended to recite that if the request is linked to *any other request*, the action of step e.) is taken. This amendment is intended to clarify that it does not matter to which request the pending request is linked, only that it is so linked to any other request. With this change, it is believed Claims 14-19 satisfy the requirements of 35 USC §112, and it is requested that this rejection be withdrawn.

3. Claims 1, 2, 4-6, 12-16, 19-24, 29-32, and 34-36 were rejected under 35 USC §102(e) as being anticipated by U.S. Patent No. 6,434,641 to Haupt et al. ("Haupt"). This rejection is respectfully traversed.

Before considering the amended claim language in detail, a summary of the Haupt system and of Applicants' invention is provided for discussion purposes. Haupt describes memory control logic within a main memory system. The control logic creates a linked list of requests that are pending to a same address within the memory system. (See Haupt Figure 2, which shows the control logic, MCA 250, and the memory 235A – 235D.) This linked list is then used to temporarily defer the submission of a request for a data item to main memory if a previous request for the same data item has already been submitted to main memory, it has been determined that this data item is not resident, and an operation to obtain that data from a cache within the system has been initiated. The subsequent (linked) request is only processed after the data item is returned to the main memory so that the request can be successfully completed. (See Haupt column 18 lines 47-59.).

It may be noted that in the Haupt system, *all of the requests* that are issued to the main memory are actually transferred to, and stored within, the main memory. Referring to the Haupt Figure 1, this means that all requests from a processing modules (POD) 120 or an I/O module 140 to main memory are submitted via the memory interfaces 130 to the Memory Storage Units (MSUs) 110. This occurs even if multiple requests are issued from the same POD that are requesting the same data. The existence of two requests from the same POD for the same data is only detected *after both* of these requests are stored within the main memory 110. This has the disadvantage of unnecessarily consuming the bandwidth of the memory interfaces 130. In addition, the MSUs are eventually forced to process both requests for the same data, even though processing for the request that was issued later will be delayed. By processing requests for the same data in this manner, "memory thrashing" may occur as data from a first processor is flushed back to the main memory before it is given to a different processor within the same POD. (See Applicants' Specification page 16, third full paragraph.)

An alternative approach to the Haupt mechanism is disclosed by Applicants' Specification. According to this new approach, the task of tracking duplicate requests is off-loaded to the processing modules (shown as processing nodes 120 in Applicants' Figure 1). That is, logic within Applicants' processing nodes, and *not within the memory,* is responsible for determining when multiple requests are being issued for the same data item by the same processing node. If this type of scenario is detected, this logic, shown as processor node director 102 and surrounding logic, creates a linked list of the requests. (See Applicants' Figures 1 and 2.) Only the first of these linked requests is transferred to SCD 100, which is the main memory of Applicants' system When this request results in the return of data from the SCD 100 to the processing node, the processor node director and other related logic of Figure 2 are responsible for ensuring that all linked requests within the processing node are handled in a manner that maintains memory coherency.

Applicants' invention provides the advantage of off-loading the handling of the multiple requests to the processor nodes. Since "duplicate" requests are no longer being issued to the main memory, but instead are maintained within the processing nodes themselves, traffic on the memory interfaces 109 is significantly reduced. (See Applicants' Figure 1 and Applicants' Specification page 17, first full paragraph.) This also eliminates the need to handle this "duplicate" request traffic within the main memory so that memory throughput is maximized. Further, the invention minimizes memory thrashing, which occurs when a processor node unnecessarily flushes data to main memory when that data is being requested by a different processor within the same processing node. (See Applicants' Specification page 33, first full paragraph.)

It may be noted that Applicants' invention is not merely an obvious modification of the system shown in Haupt. Applicants' processing nodes must handle linked requests in a manner that depends on the type of access rights granted to the processing node for a particular data item, as well as on the type of rights that are requested by a next linked request. (See, for example,

Applicants' Specification page 18, first full paragraph.) The circuitry required to handle this task is non-trivial, as described throughout Applicants' Specification. In contrast, the processing of a linked request within the Haupt system is relatively straight-forward, consisting primarily of clearing a designator for a given linked request so that the request then becomes eligible for presentation to the main memory. (See Haupt column 19 line 63 – column 20 line 8.)

Applicants' invention must further address significant coherency issues not implicated by the Haupt system. For example, Applicants' system must ensure that while linked requests are being processed, data stored within the shared cache of a POD is maintained in a coherent state. This is non-trivial, as is described, in part, on pages 17 through 22 of Applicants' Specification. Applicants' system must further handle the scenario wherein a request for data is received from the main memory by the POD while a linked list of requests for that same data is being processed by the POD. Special logic such as Remote Tracker 252 (Applicants' Figure 2) is needed to ensure that all requests are handled in a manner that maintains memory coherency. This is described, in part, on pages 24 through 27 of Applicants' Specification.

To summarize, the system of Haupt utilizes linked lists to handle requests pending within a main memory. These requests may be handled in a relatively straight-forward manner. In contrast, Applicants' system off-loads certain request handling functions to the processing nodes themselves to conserve bandwidth on the memory interfaces and within the memory itself, and to further prevent memory thrashing. This creates significant complexities which must be handled by the logic shown and described in Applicants' Specification and Drawings.

Before continuing, the above summary is intended to be consistent with Applicants' Drawings and Specification, and is not intended to modify that disclosure in any manner. With the foregoing summary available for discussion purposes, the language of the Claims is considered in more detail. Applicants' Claim 1, as currently amended, appears as follows:

1. For use in a system having multiple processors in a processing node coupled to a memory, a method, comprising:

a.) receiving multiple requests for data from the multiple processors;

b.) if ones of the multiple requests are requesting the same data, creating a respective linked list to record the ones of the multiple requests; and

c.) issuing one of the requests recorded by each linked list from the processing node to the memory.

This language describes the aspect wherein the processors are included in a processing node. If multiple requests are requesting the same data, a linked list is created to record these requests, and only one request from the linked list is issued from the processing node to the memory. In contrast, in Haupt, all of the requests are issued from the processing nodes to the memory (i.e., from processing modules 120 to MSUs 110) regardless of whether there are any pending requests for the same data. Because all requests are transferred from the processing nodes to the memory in this manner, traffic is increased on the memory interfaces 130 and memory resources are used to process the additional requests. Memory thrashing is also increased.

Once requests are resident in the Haupt memory, the requests are routed to an appropriate memory cluster control block 520 within an MSU 110. This memory cluster control block is a control circuit that may add a request to a linked list if it addresses the same data as a previous request. Eventually, the request will be forwarded from this memory cluster control block 520 to the memory cluster 235 (MCL) to which it is coupled for processing when the requested data becomes available.

As shown in Haupt Figure 6 and the corresponding description, each memory cluster control block includes combination and storage logic which

"...handles one-fourth of the address range of the MSU. The selected memory cluster control block stores the request address and control

> signals until the request is selected for presentation to the associated MCL..." (Haupt column 10 lines 43-48.)

A memory cluster control block cannot be characterized as a processor node that has multiple processors, as is claimed by Applicants' Claim 1. Thus, in Haupt, the providing of a linked request from the memory cluster control block to an MCL does not teach, or in any way suggest, Applicants' step of providing a request from a linked list within a processing node to the memory, where the processing node is defined as having multiple processors.

To summarize, in Haupt, all requests for data that is not resident within a POD itself are issued from the POD to the memory, and control logic within the memory then manages these requests, including all requests for the same data. In contrast, Applicants' method of Claim 1 off-loads the function of managing requests for the same data to each of the processing nodes. Because Haupt does not teach each and every element of Applicants' Claim 1, the rejection under 35 USC §102(e) is improper, and should be withdrawn.

Before continuing, it is reiterated that the Haupt disclosure does not even begin to suggest Applicants' invention, which requires extensive logic to solve the memory coherency issues associated with off-loading the request handling functionality from the main memory to the processing nodes.

Claims 2, and 4-6 depend from Claim 1 and include additional scopes and aspects of Applicants' invention not taught by Haupt. For at least the reasons discussed above in regards to Claim 1, Claims 2 and 4-6 are allowable over the current rejection, which should be withdrawn.

Claim 12 includes aspects similar to those discussed in Claim 1. As amended, this Claim appears as follows:

> 12. A method of processing requests from requesters to a memory, including:

a.) receiving a request for data stored in the memory;

b.) if the request is requesting the same data as another pending request that has not yet been provided from the requesters to the memory, linking the request to the other pending request; and

c.) repeating steps a.) and b.) for any additional requests issued to the memory.

Claim 12 describes linking a pending request to another pending request that requests the same data and that has not yet been provided from the requesters (i.e., processors) to the memory. This is not taught by Haupt, which describes providing all requests from the requesters to the memory, and then creating a linked list of all of the requests that have already been received by the memory and that request the same data. For this reason, Claim 12 is not taught by Haupt, and this rejection should be withdrawn.

Claims 13 -16 depend from Claim 12, include additional scopes and aspects not taught by Haupt, and are allowable over this rejection for at least the reasons discussed above in regards to Claim 12.

Claim 20, as currently amended, is as follows:

20. A system for processing requests to a memory, comprising:

multiple requesters to issue requests for data to the memory;

a request tracking circuit coupled to the multiple requesters to retain a record of each request until the request is completed, and to associate a request with any other one or more requests for the same data so that a single request for any given data is provided from the multiple requesters to the memory at a given time.

Claim 20 therefore describes a single request being provided from the multiple requesters to the memory after an association is formed between one or more

requests that are requesting the same data. In Haupt, any association is formed after all requests have been provided by the requesters (i.e., the processors) to the memory. Since Haupt does not teach each and every aspect of Applicants' amended Claim 20, this rejection should be withdrawn.

Claims 21-24 depend from Claim 20, and are allowable for at least the reasons discussed above in regards to Claim 20.

Applicants' independent Claim 29 has been amended as follows:

29. A data processing system comprising:

a memory;

a processing node coupled to the memory and having one or more requesters to generate requests for data to the memory, wherein the processing node includes a requesting tracking circuit to record, in time-order, requests issued for the same data, and to allow only one of the requests for the same data from being issued to the memory at a given time.

This Claim describes a processing node having one or more requesters to generate requests for data to the memory. Applicants' processing node includes a tracking circuit to record requests for the same data, and which allows only one of these requests to be issued to the memory. Haupt does not describe a processing node that has requesters to generate requests, and that also contains a request tracking circuit. In Haupt, the request tracking is performed within the memory cluster control blocks, which are memory control circuits within a main memory. The Haupt memory cluster control blocks do not include requesters that generate requests. Rather the memory cluster control blocks only handle *previously-generated requests* that were issued by requesters included in the PODs and I/O modules of blocks 120 and 140 of Haupt Figure 1. Thus, Claim 29 is allowable over this rejection, and should be withdrawn.

Claims 30-32 depend from Claim 29 and are allowable over this rejection for at least the reasons discussed above in regards to Claim 29.

Claim 34 has been amended as follows:

34. A system for processing requests to a memory, including: processing means for originating the requests to the memory; and request tracking means for receiving the requests, and for forming an association between any of the requests that are requesting the same data, and for allowing only one of the associated requests to be provided from the processing means to the memory.

According to Claim 34, the processing means is for originating requests, and the request tracking means is for forming an association between requests requesting the same data. The request tracking means is further for allowing only one of the associated requests from being provided from the processing means (which originates the requests) and the memory. As previously discussed, in Haupt, all requests are provided from the processing means that originates the requests (i.e., the POD or I/O Module) to the memory. It is only *after* the requests reside within the memory that associations are created between requests so that the requests can be processed. Moreover, for reasons discussed above, the memory cluster control means that forms the linked lists in Haupt cannot be characterized as processing means that *originates* requests, since this logic only routes existing requests. Thus, Haupt does not teach the system of Applicants' Claim 34.

Claims 35 and 36 depend from Claim 34 and are allowable over this rejection for at least the reasons set forth in regards to Claim 34.

To summarize, the amendments to the Claims clarify the distinctions between the Haupt system and that of Applicants' invention. With these

changes, Haupt neither teaches the claimed invention, nor renders that invention obvious. For this reason, it is believed all pending Claims are now in condition for allowance.

4. The indication of allowable subject matter in Claims 3, 7-11, 17-18, 25-28, 33 and 37 is appreciatively acknowledged. In view of the amendments set forth above with respect to the corresponding independent Claims, it is believed all of these dependent Claims are now in condition for allowance.

5. The prior art made of record and not relied upon has been reviewed and is considered to be of general interest only.

Conclusion

In the Office Action dated July 15, 2005, Claims 1, 2, 4-6, 12-24, 29-32 and 34-36 were rejected and Claims 3, 7-11, 25-28, 33 and 37 were objected to. In the amendment set forth above, Claims 1, 12, 14, 20, 24, 29, and 34 are amended, and the remaining Claims are as previously presented. In view of the amendments to the Claims and the comments set forth above, it is respectfully submitted that all Claims are now in condition for allowance as presently presented, and a Notice of Allowance is therefore respectfully requested.

Respectfully submitted,

Beth L McMahon 9/12/2005

Beth L. McMahon
Attorney for Applicants
Reg. No. 41,987
Tele No. (651) 635-7893

Unisys Corporation
M.S. 4773
P.O. Box 64942
St. Paul, MN 55164-0942